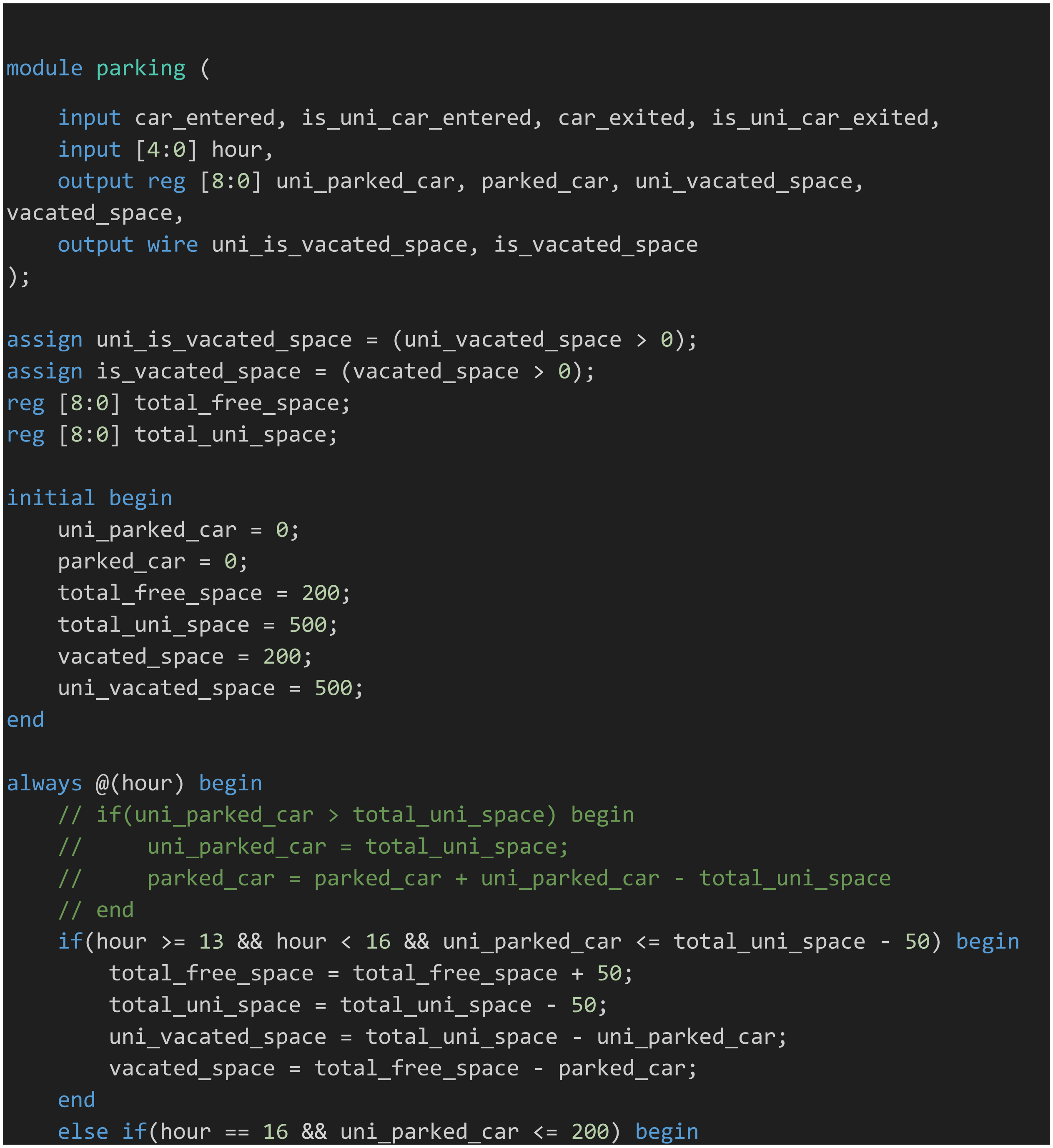
به نام خدا

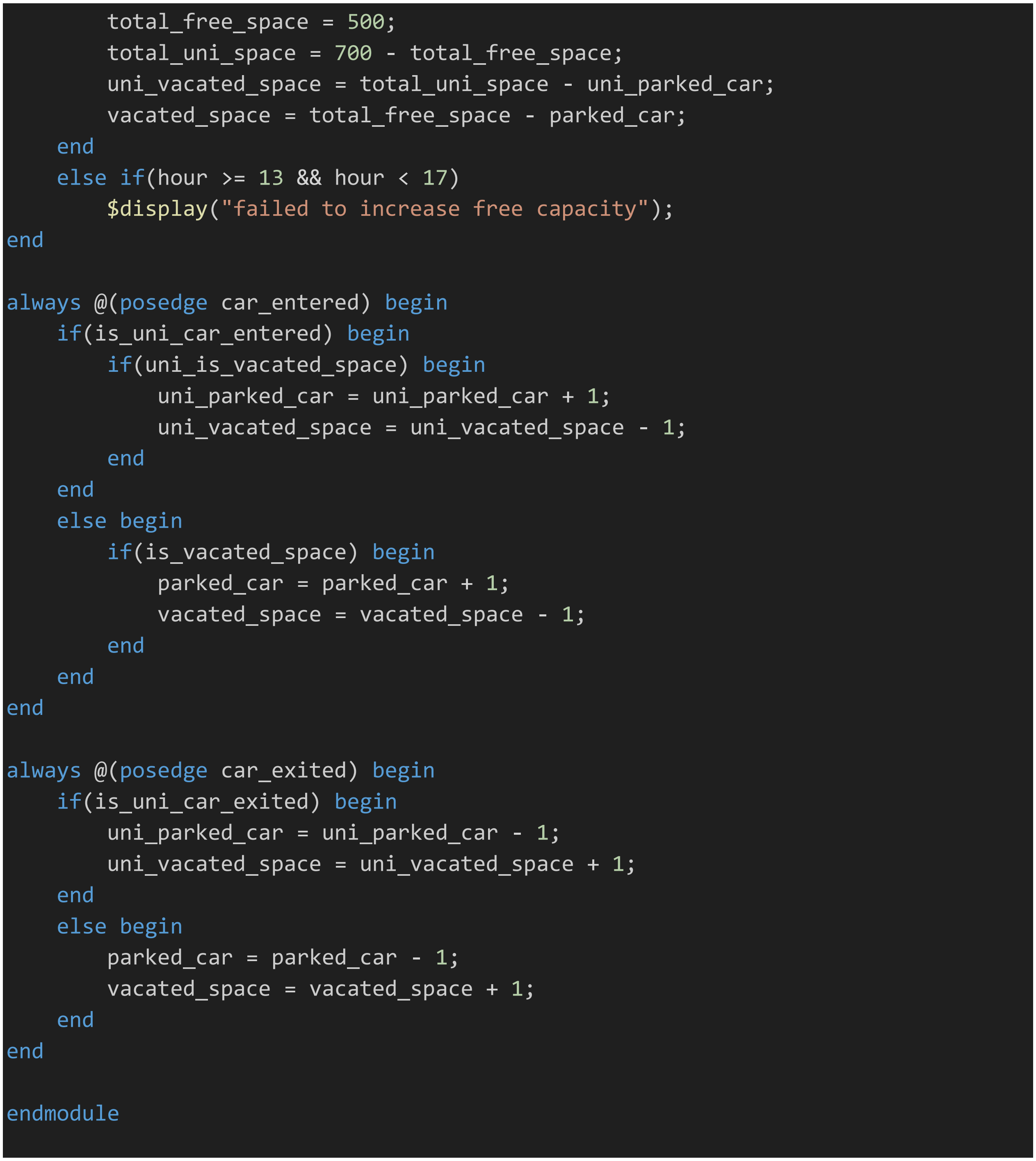
سید مبین رضوی 401110267

سوال 8 – الف)

کد کامل سوال 8 را میتوانید در زیر ببینید :



```
module parking (
    input car_entered, is_uni_car_entered, car_exited, is_uni_car_exited,
    input [4:0] hour,
    output reg [8:0] uni_parked_car, parked_car, uni_vacated_space,
vacated_space,
    output wire uni_is_vacated_space, is_vacated_space
);

assign uni_is_vacated_space = (uni_vacated_space > 0);
assign is_vacated_space = (vacated_space > 0);
reg [8:0] total_free_space;
reg [8:0] total_uni_space;

initial begin
    uni_parked_car = 0;
    parked_car = 0;
    total_free_space = 200;
    total_uni_space = 500;
    vacated_space = 200;
    uni_vacated_space = 500;
end

always @(hour) begin
    // if(uni_parked_car > total_uni_space) begin
    //     uni_parked_car = total_uni_space;
    //     parked_car = parked_car + uni_parked_car - total_uni_space
    // end
    if(hour >= 13 && hour < 16 && uni_parked_car <= total_uni_space - 50) begin
        total_free_space = total_free_space + 50;
        total_uni_space = total_uni_space - 50;
        uni_vacated_space = total_uni_space - uni_parked_car;
        vacated_space = total_free_space - parked_car;
    end
    else if(hour == 16 && uni_parked_car <= 200) begin
```



```
        total_free_space = 500;
        total_uni_space = 700 - total_free_space;
        uni_vacated_space = total_uni_space - uni_parked_car;
        vacated_space = total_free_space - parked_car;
    end
    else if(hour >= 13 && hour < 17)
        $display("failed to increase free capacity");
end

always @(posedge car_entered) begin
    if(is_uni_car_entered) begin
        if(uni_is_vacated_space) begin
            uni_parked_car = uni_parked_car + 1;
            uni_vacated_space = uni_vacated_space - 1;
        end
    end
    else begin
        if(is_vacated_space) begin
            parked_car = parked_car + 1;
            vacated_space = vacated_space - 1;
        end
    end
end

always @(posedge car_exited) begin
    if(is_uni_car_exited) begin
        uni_parked_car = uni_parked_car - 1;
        uni_vacated_space = uni_vacated_space + 1;
    end
    else begin
        parked_car = parked_car - 1;
        vacated_space = vacated_space + 1;
    end
end

endmodule
```

چند نکته در رابطه با کد :

1) از آنجاکه در صورت سوال ذکر نشده بود ، از این فرض منطقی استفاده کردم که اگر ظرفیت ماشین های دانشگاهی کمتر از 50 تا جایگاه خالی

داشته باشد نتوان در ساعات ذکر شده ، ظرفیت آزاد را 50 تا افزایش داد (زیرا ظرفیت کل ثابت است و برای اضافه کردن به ظرفیت آزاد باید از ظرفیت دانشگاهی کم کرد) و ارور زیر داده شود :

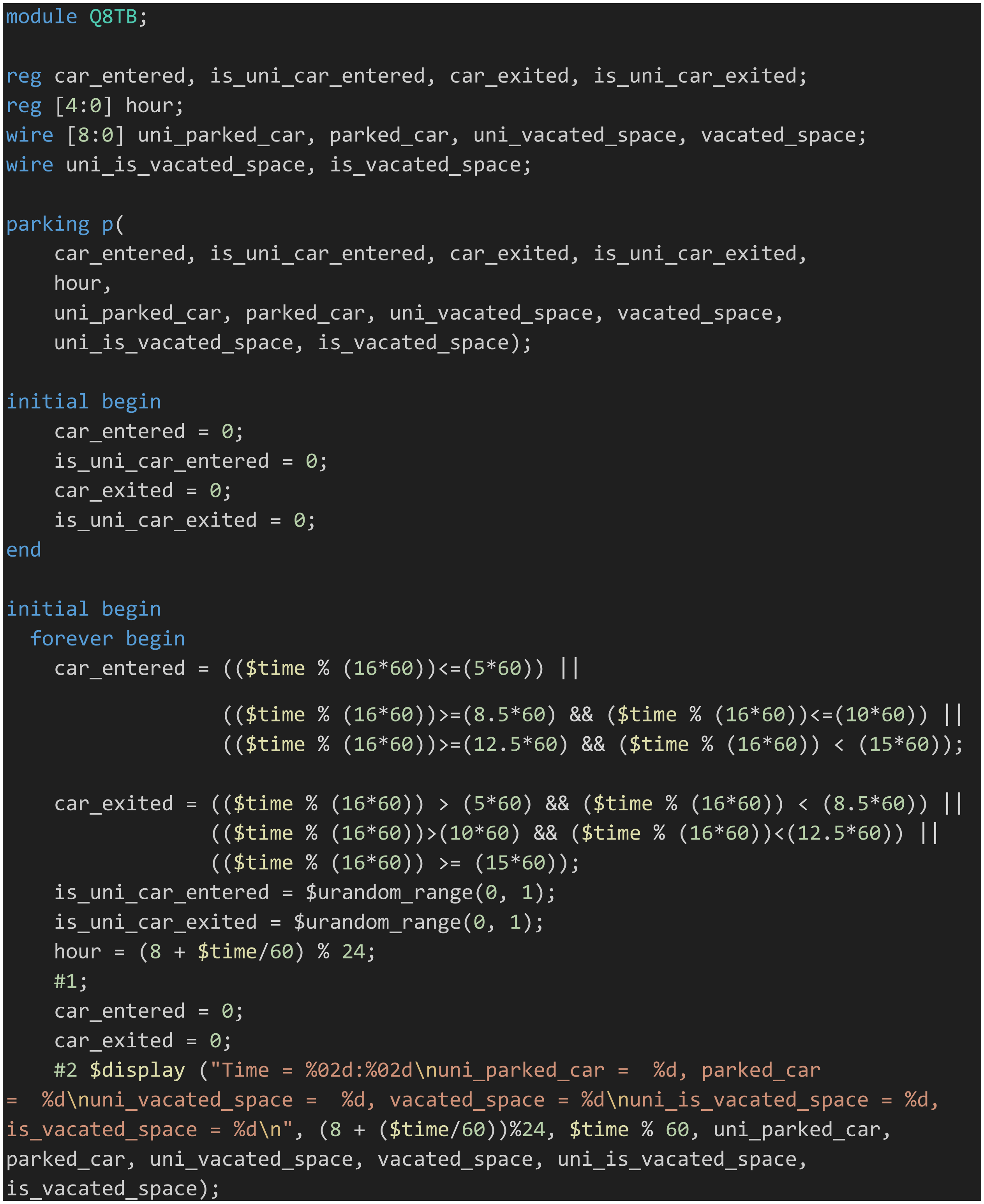
Failed to increase free capacity

2) در صورتیکه ظرفیت دانشگاهی پر باشد و ماشین دانشگاهی بخواهد وارد بشود ، به آن اجازه ی ورود داده نخواهد شد به معنی که حتی به عنوان ماشین آزاد نیز نمیتواند وارد بشود.

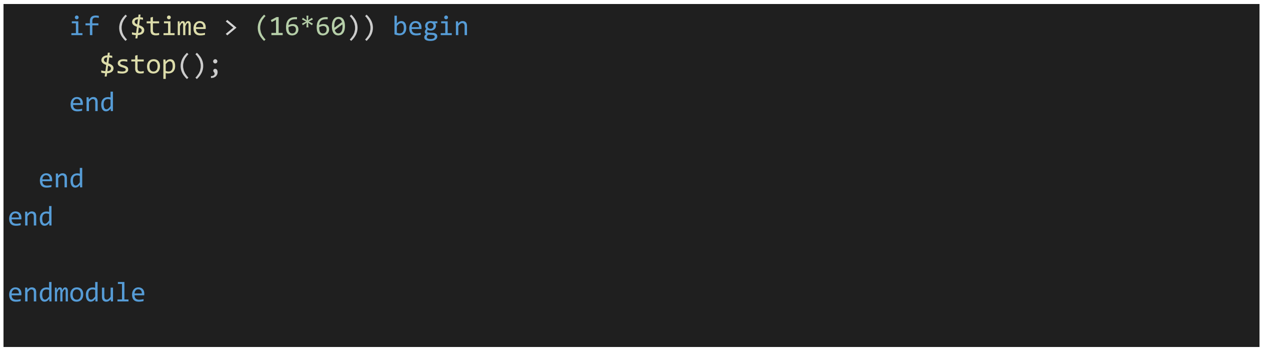
مورد اول و به طور کلی تغییر ظرفیت ها راس ساعات خاص ، داخل بلاک always @(hour) begin انجام می شود. مورد دوم و به طور کلی ورود ماشین ها نیز در بلاک always @(posedge car_entered) begin انجام میشود.

در زیر نیز می توانید تست بنچ را مشاهده کنید. در اینجا هر یک واحد از گذر متغیر time معادل یک دقیقه است ، بنابراین 960 دقیقه معادل 16 ساعت میشود که به نوعی از ساعت 8 الی 24 محسوب خواهد شد. همچنین هر 3 دقیقه یا یک ماشین قصد ورود میکند و یا یک ماشین قصد خروج میکند.

در تست بنچ زیر ، در ساعات 8 تا 13 و 16.5 تا 18 و 20.5 الی 23 ماشین ها وارد می شوند (نوع ماشین ورودی به صورت یکنواخت انتخاب می شود) و در باقی ساعات ماشین ها خارج می شوند.

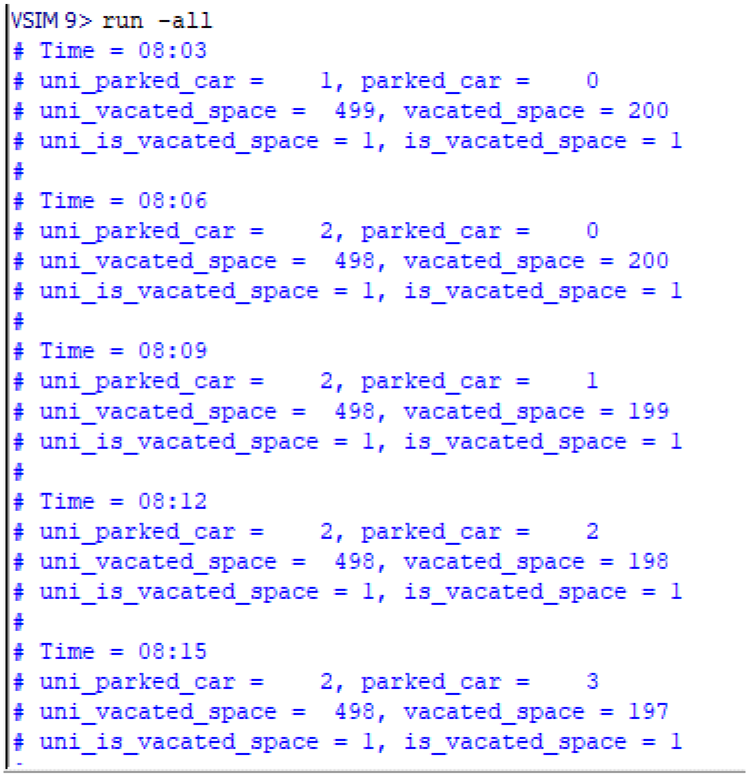


```
module Q8TB;

reg car_entered, is_uni_car_entered, car_exited, is_uni_car_exited;
reg [4:0] hour;
wire [8:0] uni_parked_car, parked_car, uni_vacated_space, vacated_space;
wire uni_is_vacated_space, is_vacated_space;

parking p(
    car_entered, is_uni_car_entered, car_exited, is_uni_car_exited,
    hour,
    uni_parked_car, parked_car, uni_vacated_space, vacated_space,
    uni_is_vacated_space, is_vacated_space);

initial begin
    car_entered = 0;
    is_uni_car_entered = 0;
    car_exited = 0;
    is_uni_car_exited = 0;
end

initial begin
  forever begin
    car_entered = (($time % (16*60))<=(5*60)) ||
                  (($time % (16*60))>=(8.5*60) && ($time % (16*60))<=(10*60)) ||
                  (($time % (16*60))>=(12.5*60) && ($time % (16*60)) < (15*60));

    car_exited = (($time % (16*60)) > (5*60) && ($time % (16*60)) < (8.5*60)) ||
                 (($time % (16*60))>(10*60) && ($time % (16*60))<(12.5*60)) ||
                 (($time % (16*60)) >= (15*60));
    is_uni_car_entered = $urandom_range(0, 1);
    is_uni_car_exited = $urandom_range(0, 1);
    hour = (8 + $time/60) % 24;
    #1;
    car_entered = 0;
    car_exited = 0;
    #2 $display ("Time = %02d:%02d\nuni_parked_car =  %d, parked_car
=  %d\nuni_vacated_space =  %d, vacated_space = %d\nuni_is_vacated_space = %d,
is_vacated_space = %d\n", (8 + ($time/60))%24, $time % 60, uni_parked_car,
parked_car, uni_vacated_space, vacated_space, uni_is_vacated_space,
is_vacated_space);
```

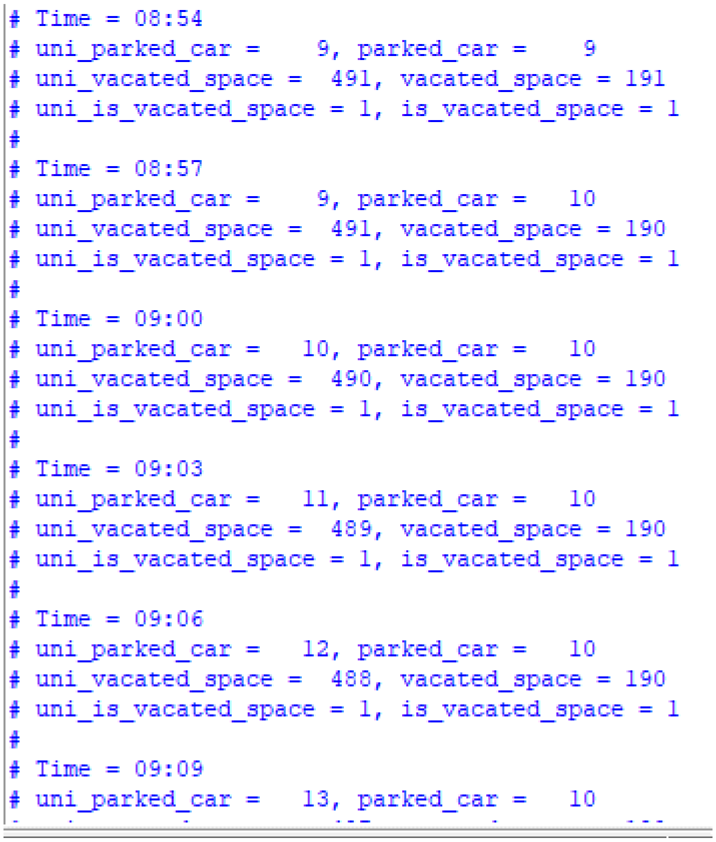


```
    if ($time > (16*60)) begin
      $stop();
    end

  end
end

endmodule
```

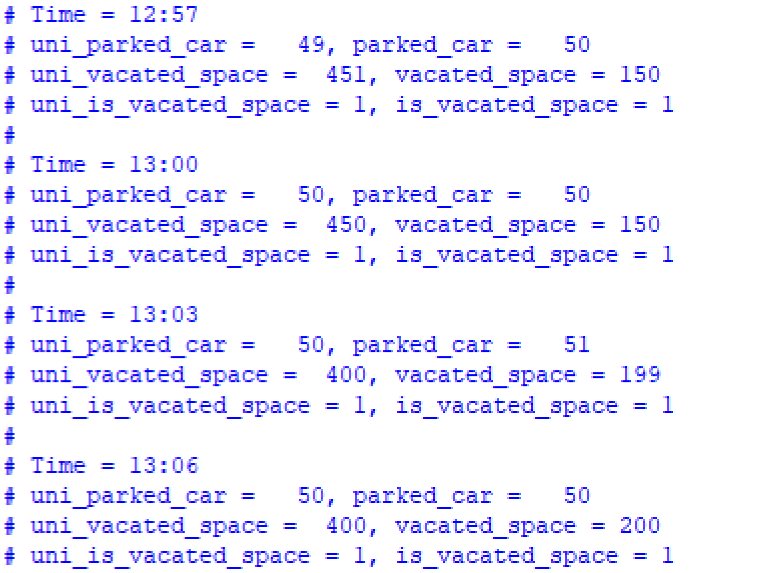
در زیر چند عکس از نتیجه ی خروجی آن را مشاهده می کنید :



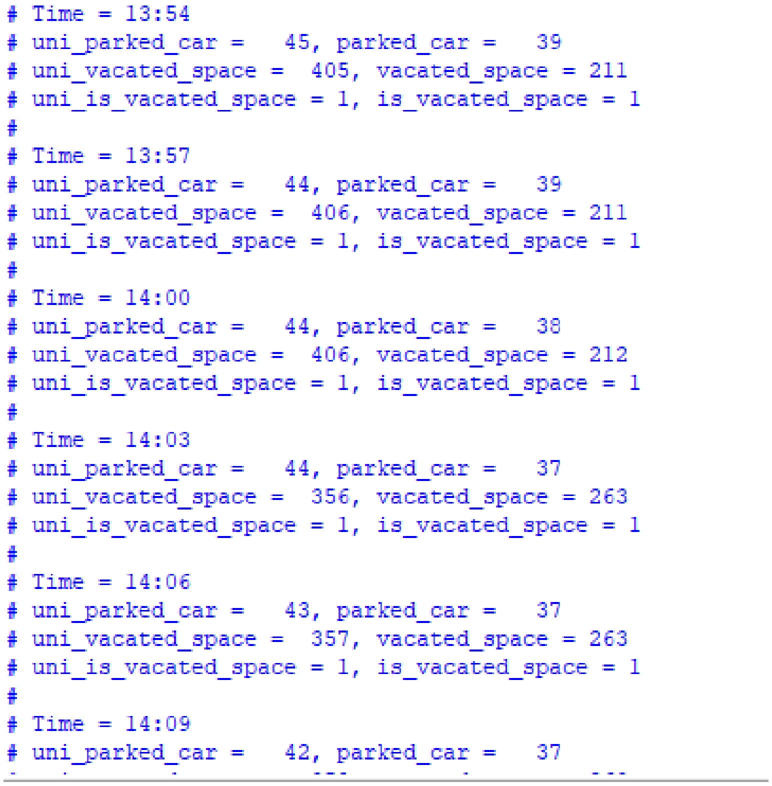
```
VSIM 9> run -all
# Time = 08:03
# uni_parked_car =     1, parked_car =     0
# uni_vacated_space =  499, vacated_space = 200
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 08:06
# uni_parked_car =     2, parked_car =     0
# uni_vacated_space =  498, vacated_space = 200
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 08:09
# uni_parked_car =     2, parked_car =     1
# uni_vacated_space =  498, vacated_space = 199
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 08:12
# uni_parked_car =     2, parked_car =     2
# uni_vacated_space =  498, vacated_space = 198
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 08:15
# uni_parked_car =     2, parked_car =     3
# uni_vacated_space =  498, vacated_space = 197
# uni_is_vacated_space = 1, is_vacated_space = 1
```



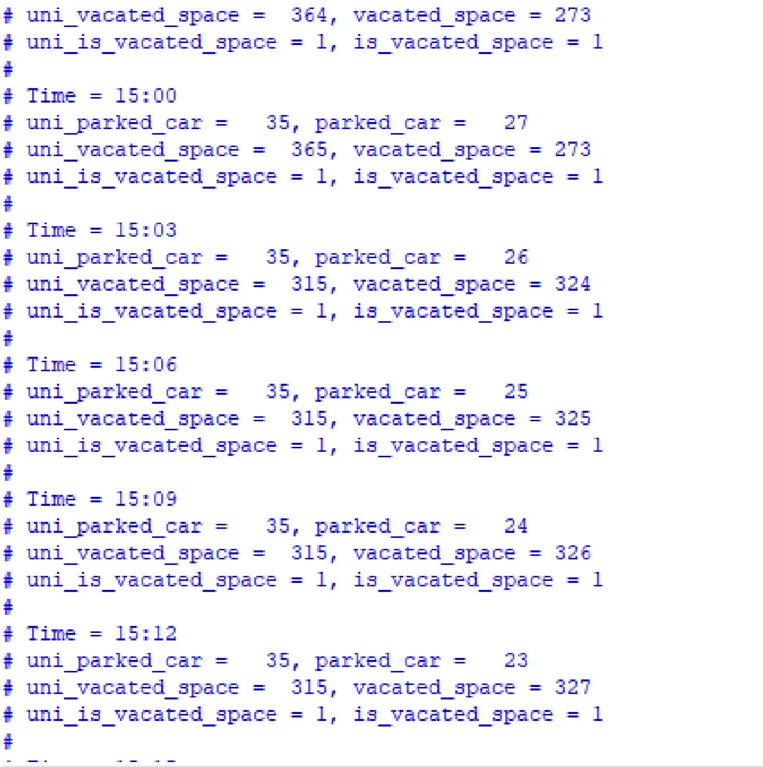
```
# Time = 08:54
# uni_parked_car =     9, parked_car =     9
# uni_vacated_space =  491, vacated_space = 191
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 08:57
# uni_parked_car =     9, parked_car =    10
# uni_vacated_space =  491, vacated_space = 190
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 09:00
# uni_parked_car =    10, parked_car =    10
# uni_vacated_space =  490, vacated_space = 190
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 09:03
# uni_parked_car =    11, parked_car =    10
# uni_vacated_space =  489, vacated_space = 190
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 09:06
# uni_parked_car =    12, parked_car =    10
# uni_vacated_space =  488, vacated_space = 190
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 09:09
# uni_parked_car =    13, parked_car =    10
```



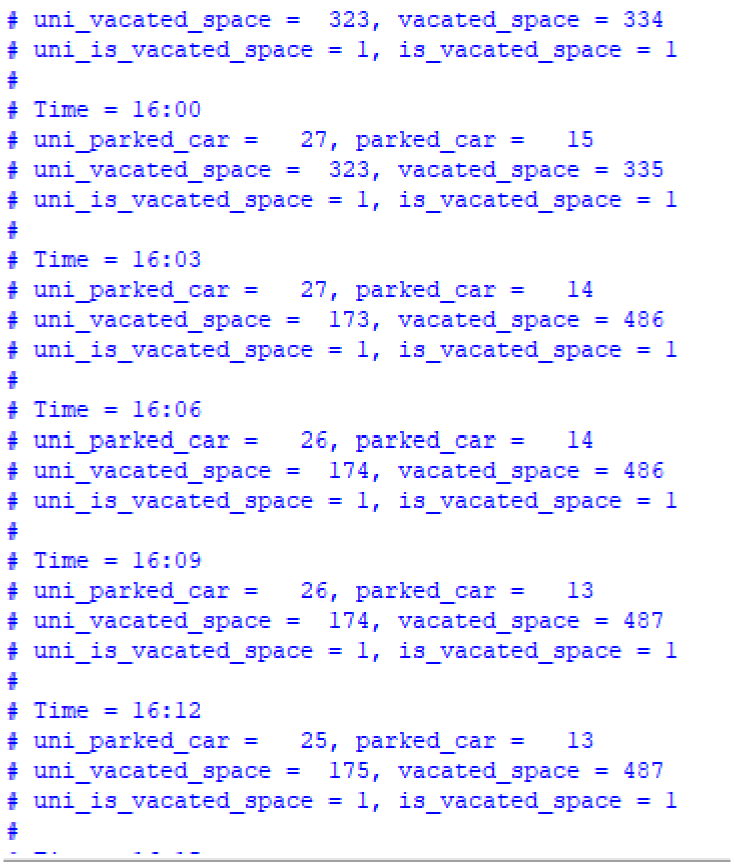
```
# Time = 12:57
# uni_parked_car =    49, parked_car =    50
# uni_vacated_space =  451, vacated_space = 150
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 13:00
# uni_parked_car =    50, parked_car =    50
# uni_vacated_space =  450, vacated_space = 150
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 13:03
# uni_parked_car =    50, parked_car =    51
# uni_vacated_space =  400, vacated_space = 199
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 13:06
# uni_parked_car =    50, parked_car =    50
# uni_vacated_space =  400, vacated_space = 200
# uni_is_vacated_space = 1, is_vacated_space = 1
```



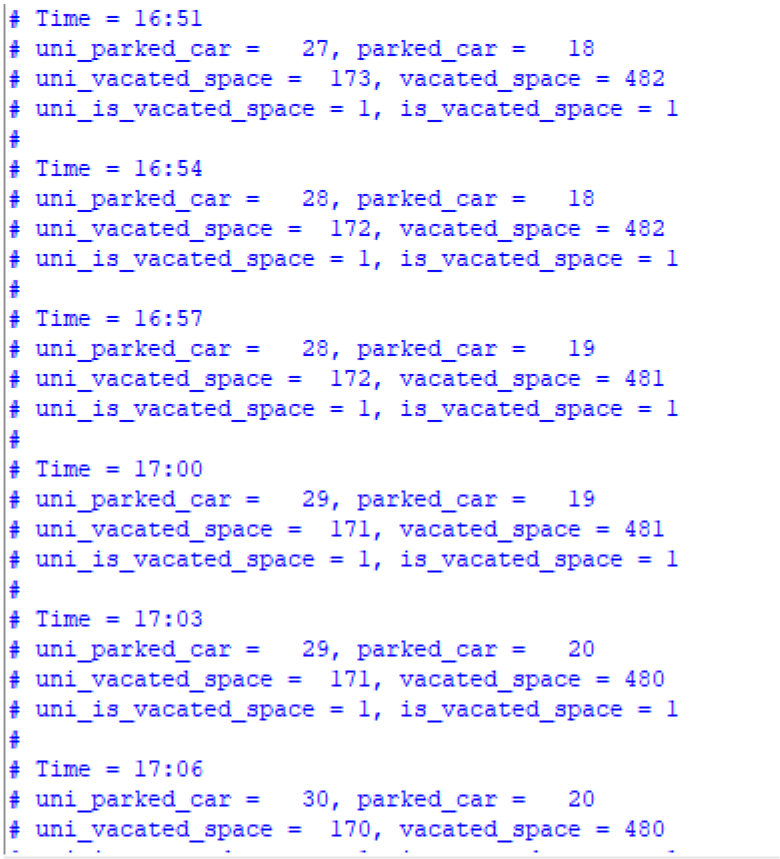
```
# Time = 13:54
# uni_parked_car =   45, parked_car =   39
# uni_vacated_space =  405, vacated_space = 211
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 13:57
# uni_parked_car =   44, parked_car =   39
# uni_vacated_space =  406, vacated_space = 211
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 14:00
# uni_parked_car =   44, parked_car =   38
# uni_vacated_space =  406, vacated_space = 212
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 14:03
# uni_parked_car =   44, parked_car =   37
# uni_vacated_space =  356, vacated_space = 263
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 14:06
# uni_parked_car =   43, parked_car =   37
# uni_vacated_space =  357, vacated_space = 263
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 14:09
# uni_parked_car =   42, parked_car =   37
```



```
# uni_vacated_space =  364, vacated_space = 273
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:00
# uni_parked_car =   35, parked_car =   27
# uni_vacated_space =  365, vacated_space = 273
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:03
# uni_parked_car =   35, parked_car =   26
# uni_vacated_space =  315, vacated_space = 324
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:06
# uni_parked_car =   35, parked_car =   25
# uni_vacated_space =  315, vacated_space = 325
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:09
# uni_parked_car =   35, parked_car =   24
# uni_vacated_space =  315, vacated_space = 326
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:12
# uni_parked_car =   35, parked_car =   23
# uni_vacated_space =  315, vacated_space = 327
# uni_is_vacated_space = 1, is_vacated_space = 1
#
```



```
# uni_vacated_space =  323, vacated_space = 334
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:00
# uni_parked_car =   27, parked_car =   15
# uni_vacated_space =  323, vacated_space = 335
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:03
# uni_parked_car =   27, parked_car =   14
# uni_vacated_space =  173, vacated_space = 486
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:06
# uni_parked_car =   26, parked_car =   14
# uni_vacated_space =  174, vacated_space = 486
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:09
# uni_parked_car =   26, parked_car =   13
# uni_vacated_space =  174, vacated_space = 487
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:12
# uni_parked_car =   25, parked_car =   13
# uni_vacated_space =  175, vacated_space = 487
# uni_is_vacated_space = 1, is_vacated_space = 1
#
```



```
# Time = 16:51
# uni_parked_car =   27, parked_car =   18
# uni_vacated_space =  173, vacated_space = 482
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:54
# uni_parked_car =   28, parked_car =   18
# uni_vacated_space =  172, vacated_space = 482
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 16:57
# uni_parked_car =   28, parked_car =   19
# uni_vacated_space =  172, vacated_space = 481
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 17:00
# uni_parked_car =   29, parked_car =   19
# uni_vacated_space =  171, vacated_space = 481
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 17:03
# uni_parked_car =   29, parked_car =   20
# uni_vacated_space =  171, vacated_space = 480
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 17:06
# uni_parked_car =   30, parked_car =   20
# uni_vacated_space =  170, vacated_space = 480
```

بسادگی می توانید افزایش ظرفیت آزاد را در ساعات 13:03 و 14:03 و 15:03 و 16:03 مشاهده کنید .

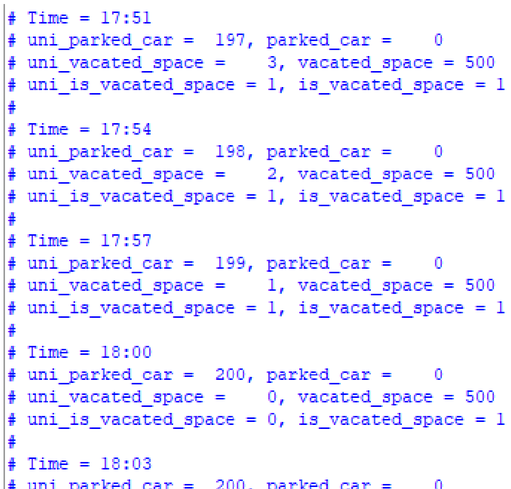
اکنون حالت دیگری را تست میکنیم که اگر ظرفیت پر باشد ، به ماشین هایی که میخواهند وارد شوند اجازه ی ورود داده نشود :

در تست بنچ قبل ، صرفا داخل forever begin را با کد زیر تعویض میکنیم :

```
    car_entered = ($time <= (14*60));
    car_exited = ($time > (14*60));
    is_uni_car_entered = 1;
    is_uni_car_exited = $urandom_range(0, 1);
    hour = (8 + $time/60) % 24;
    #1;
    car_entered = 0;
    car_exited = 0;
    #2 $display ("Time = %02d:%02d\nuni_parked_car =  %d, parked_car
=  %d\nuni_vacated_space =  %d, vacated_space = %d\nuni_is_vacated_space = %d,
is_vacated_space = %d\n", (8 + ($time/60))%24, $time % 60, uni_parked_car,
parked_car, uni_vacated_space, vacated_space, uni_is_vacated_space,
is_vacated_space);
    if ($time > (16*60)) begin
      $stop();
    end
```

در این تست بنچ ، از ساعت 8 الی 22 ، تنها ماشین های دانشگاهی در حال ورود هستند.

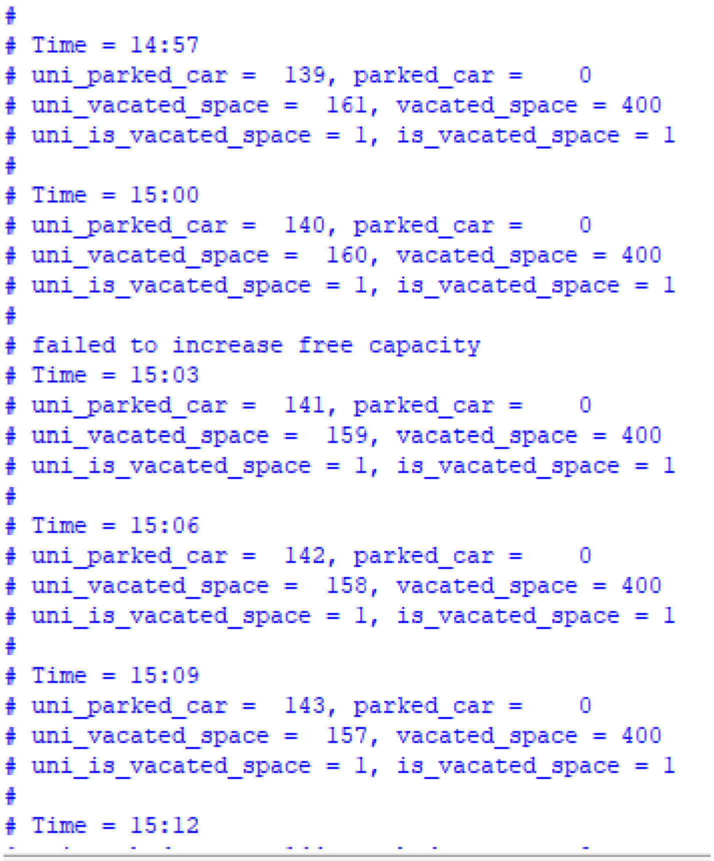
در عکس زیر نیز میتوانید مشاهده کنید که وقتی ظرفیت ماشین های دانشگاهی پر می شود دیگر ماشینی وارد پارکینگ نخواهد شد.



```
# Time = 17:51
# uni_parked_car =  197, parked_car =     0
# uni_vacated_space =     3, vacated_space = 500
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 17:54
# uni_parked_car =  198, parked_car =     0
# uni_vacated_space =     2, vacated_space = 500
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 17:57
# uni_parked_car =  199, parked_car =     0
# uni_vacated_space =     1, vacated_space = 500
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 18:00
# uni_parked_car =  200, parked_car =     0
# uni_vacated_space =     0, vacated_space = 500
# uni_is_vacated_space = 0, is_vacated_space = 1
#
# Time = 18:03
# uni_parked_car =  200, parked_car =     0
```

اکنون حالتی را تست میکنیم که با ارور failed to increase capacity مواجه بشویم : برای تست این موضوع از همان تست بنچ دوم استفاده میکنم اما در کد اصلی این فرض را میکنم که راس ساعات 13 و 14 و 15 ظرفیت آزاد به مقدار 200 واحد افزایش یابد (بجای 50 واحد) . بدین ترتیب اگر در این ساعات ، ظرفیت دانشگاهی خالی ، کمتر از 200 بود ظرفیت آزاد افزایش نخواهد یافت و ارور مربوطه نشان داده خواهد شد :

در زیر میتوانید نتیجه ی تست را ببینید :

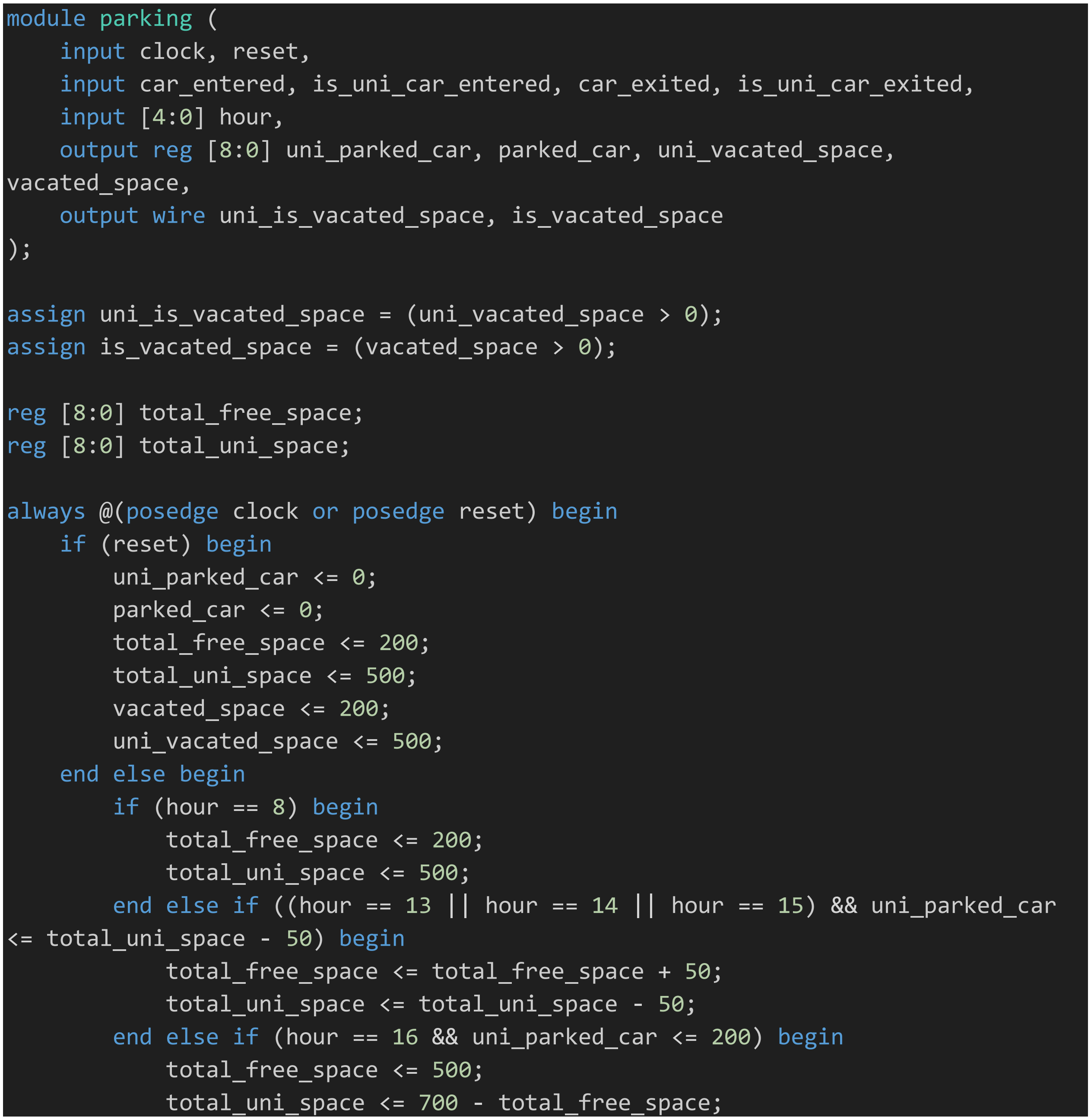


```
#
# Time = 14:57
# uni_parked_car =  139, parked_car =    0
# uni_vacated_space =  161, vacated_space = 400
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:00
# uni_parked_car =  140, parked_car =    0
# uni_vacated_space =  160, vacated_space = 400
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# failed to increase free capacity
# Time = 15:03
# uni_parked_car =  141, parked_car =    0
# uni_vacated_space =  159, vacated_space = 400
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:06
# uni_parked_car =  142, parked_car =    0
# uni_vacated_space =  158, vacated_space = 400
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:09
# uni_parked_car =  143, parked_car =    0
# uni_vacated_space =  157, vacated_space = 400
# uni_is_vacated_space = 1, is_vacated_space = 1
#
# Time = 15:12
```

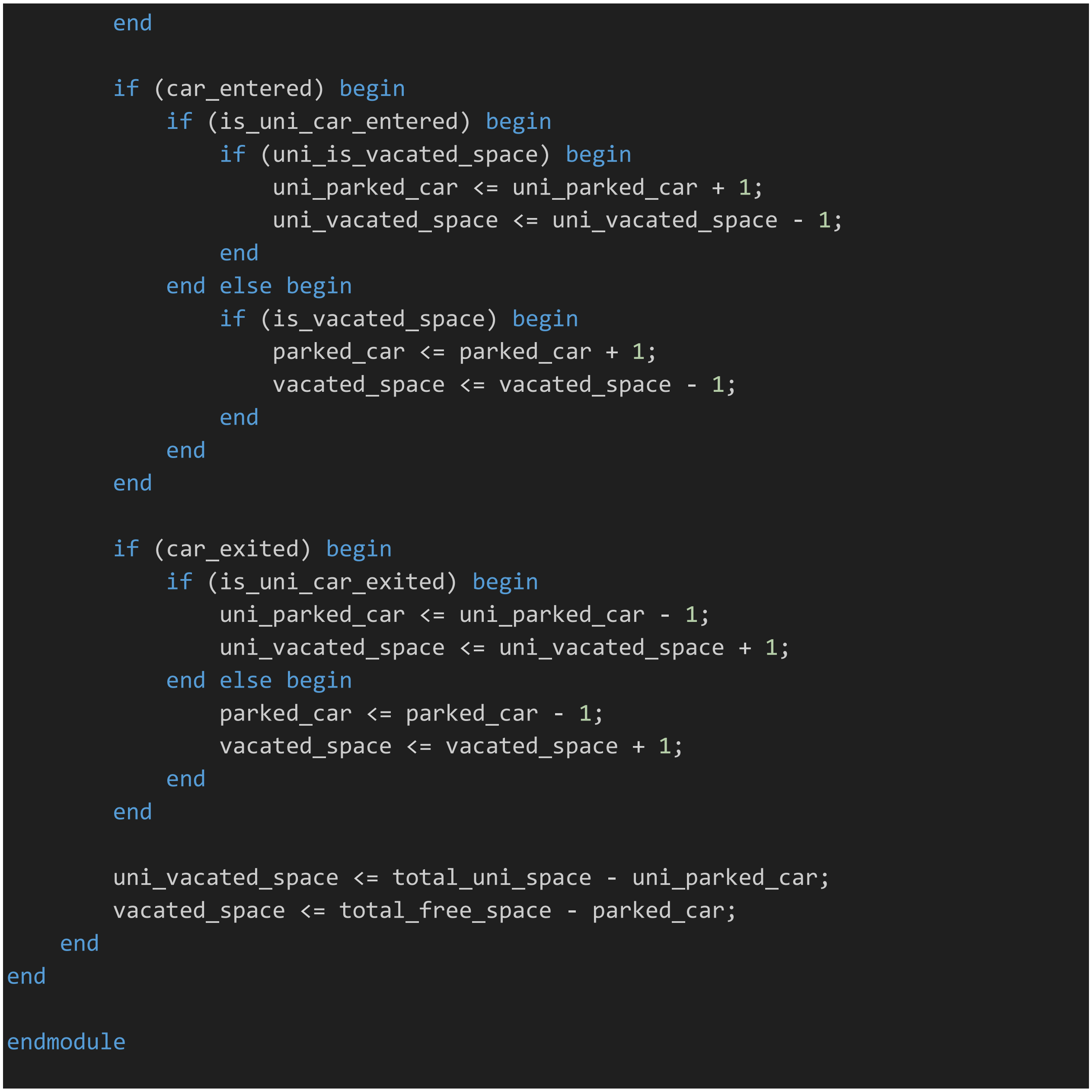
همانطور که میبینید ، در ساعت 15:03 ارور ذکرشده داده شده است زیرا ظرفیت دانشگاهی آزاد یا همان Uni_vacated_space کمتر از 200 بود پس نمیتوانست ظرفیت آزاد را افزایش دهد.

سوال 8 – ب)

در اینجا ابتدا باید کد ماژول اصلی را به یک کد قابل سنتز تبدیل کنیم. به عنوان مثال ، برای اینکه کدمان قابل سنتز باشد نباید هیچ reg ای داخل دو تا always بلاک ، آپدیت شود. کد قابل سنتز ماژول اصلی را در زیر آورده ام (دقت کنید که در کد های آپلود شده در گیتهاب ، نام آن synthesizable_Q8 می باشد .) :



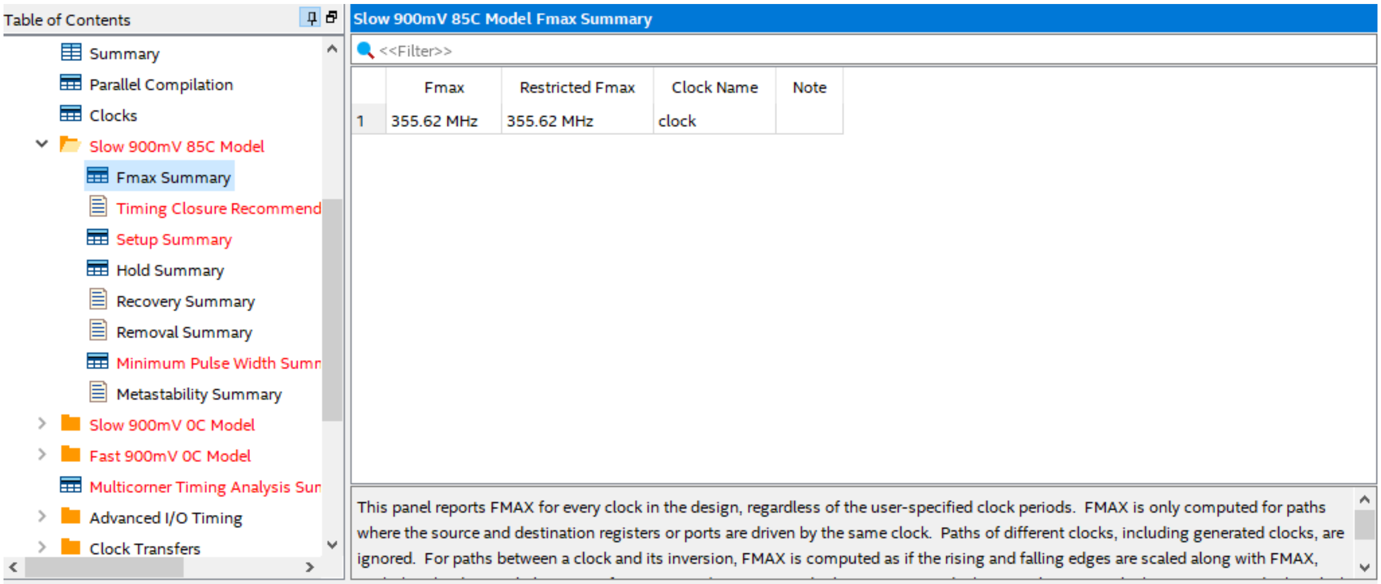
```
module parking (
    input clock, reset,
    input car_entered, is_uni_car_entered, car_exited, is_uni_car_exited,
    input [4:0] hour,
    output reg [8:0] uni_parked_car, parked_car, uni_vacated_space,
vacated_space,
    output wire uni_is_vacated_space, is_vacated_space
);

assign uni_is_vacated_space = (uni_vacated_space > 0);
assign is_vacated_space = (vacated_space > 0);

reg [8:0] total_free_space;
reg [8:0] total_uni_space;

always @(posedge clock or posedge reset) begin
    if (reset) begin
        uni_parked_car <= 0;
        parked_car <= 0;
        total_free_space <= 200;
        total_uni_space <= 500;
        vacated_space <= 200;
        uni_vacated_space <= 500;
    end else begin
        if (hour == 8) begin
            total_free_space <= 200;
            total_uni_space <= 500;
        end else if ((hour == 13 || hour == 14 || hour == 15) && uni_parked_car
<= total_uni_space - 50) begin
            total_free_space <= total_free_space + 50;
            total_uni_space <= total_uni_space - 50;
        end else if (hour == 16 && uni_parked_car <= 200) begin
            total_free_space <= 500;
            total_uni_space <= 700 - total_free_space;
```



```
        end

        if (car_entered) begin
            if (is_uni_car_entered) begin
                if (uni_is_vacated_space) begin
                    uni_parked_car <= uni_parked_car + 1;
                    uni_vacated_space <= uni_vacated_space - 1;
                end
            end else begin
                if (is_vacated_space) begin
                    parked_car <= parked_car + 1;
                    vacated_space <= vacated_space - 1;
                end
            end
        end

        if (car_exited) begin
            if (is_uni_car_exited) begin
                uni_parked_car <= uni_parked_car - 1;
                uni_vacated_space <= uni_vacated_space + 1;
            end else begin
                parked_car <= parked_car - 1;
                vacated_space <= vacated_space + 1;
            end
        end

        uni_vacated_space <= total_uni_space - uni_parked_car;
        vacated_space <= total_free_space - parked_car;
    end
end

endmodule
```

اکنون برای حساب کردن بیشترین فرکانس ممکن یا همان fmax از report های بخش سنتز کوارتوس ، در بخش timing analysis وارد بخش

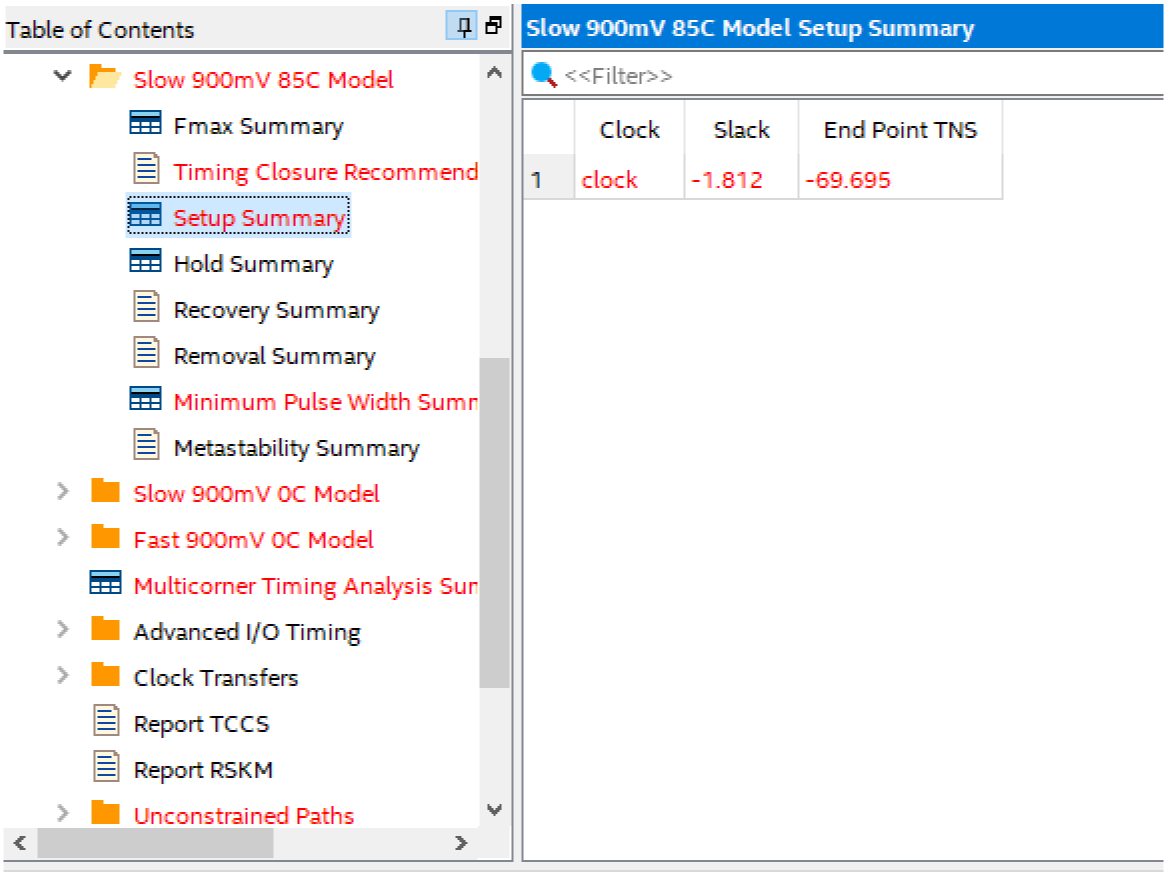
Slow 900mV 85C Model می شویم. در زیر میتوانید عکس ریپورت های fmax و مقدار slack برای setup summary را ببینید :



Slow 900mV 85C Model Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	355.62 MHz	355.62 MHz	clock	

This panel reports FMAX for every clock in the design, regardless of the user-specified clock periods. FMAX is only computed for paths where the source and destination registers or ports are driven by the same clock. Paths of different clocks, including generated clocks, are ignored. For paths between a clock and its inversion, FMAX is computed as if the rising and falling edges are scaled along with FMAX,



Slow 900mV 85C Model Setup Summary

	Clock	Slack	End Point TNS
1	clock	-1.812	-69.695

همانطور که مشاهده می کنید ، مقدار slack در بخش setup summary برابر -1.812ns می باشد . حال در مدل اصلی ، پریود کلاک برابر 1ns در نظر گرفته شده پس

$$newClockPeriod = 1ns + 1.812ns = 2.812ns$$

$$\Rightarrow maximum\ frequency = \frac{1}{2.812ns} = 0.355618GHz$$
$$= 355.62MHz$$

پس همانطور که میبینید ، مقدار بدست آمده برای fmax با مقدار دیده شده در عکس اول همخوانی دارد.